基于句法分析及主题分布的关键词抽取模型

王 昊，刘 丹†，刘 硕

(电子科技大学 电子科学技术研究院，成都 611731)

摘 要：针对 TextRank 算法在抽取篇章关键词时忽略句法信息、主题信息等问题，提出基于句法分析与主题分布的篇章关键词抽取模型。模型分为段落和篇章两阶段递进抽取篇章关键词。首先以段落为单位，结合词共现、语法及语义信息抽取段落关键词；然后根据段落主题对段落聚类，形成段落主题集；最后根据段落主题分布特征抽取篇章关键词。在公开的新闻数据集上，模型的抽取效果较原始 TextRank 提升了约 10%。实验结果表明，方法的抽取效果有了明显提升，证明了语法信息及主题信息的重要性。

关键词：关键词抽取；TextRank；依存关系；语义距离；段落主题

中图分类号：TP391　　doi: 10.19734/j.issn.1001-3695.2022.02.0068

Keyword extraction model based on syntactic analysis and topic distribution

Wang Hao, Liu Dan†, Liu Shuo

(*Research Institute of Electronic Science & Technology*, *University of Electronic Science & Technology of China*, *Chengdu* 611731, *China*)

Abstract: Aiming at the problem that TextRank ignores syntactic information and topic information when extracting chapter keywords, propose a chapter keyword extraction model based on syntactic analysis and topic distribution. Model includes two stages of chapter keyword extraction: paragraph and chapter. Firstly, use paragraphs as a unit to extract paragraph keywords by combining word co-occurrence, grammatical and semantic information. Then cluster the paragraphs according to the paragraph topics to form the paragraph topic set. Finally, extract chapter keywords based on the distribution characteristics of paragraph topics. On the open news dataset, the model's extraction effect improves by about 10% compared with the original TextRank. Results show that the method has significantly improved the extraction effect, and prove the importance of grammatical information and topic information.

Key words: keyword extraction; TextRank; dependency grammar; semantic distance; paragraph topic

0 引言

关键词是篇章内容的高度概括、主题的简明表达。关键词抽取技术在工业中有着广泛运用，其中无监督方法凭借其泛用性备受青睐。TextRank 是最具代表性的基于图的无监督抽取算法，其以词为节点构建词图，通过计算节点权重抽取关键词。但其忽略了词的语义语法信息及文本的主题信息，对长文本、多主题文本抽取效果不佳。

本文提出基于句法分析与篇章主题的关键词抽取模型 S-TAKE(Syntactic analysis and Paragraph Topic based Article Keyword Extraction Model)。模型以段落为抽取关键词的基本文本单元，由段落至篇章分两阶段抽取篇章关键词。抽取段落关键词时，利用句法分析在词图中引入语法信息，解决 TextRank 过度偏向高频词的问题；利用词嵌入在转移矩阵中引入语义信息，解决 TextRank 忽略词语义关联的问题；以段落作为抽取关键词的基本单位，解决 TextRank 对长文本处理困难的问题。筛选篇章关键词时引入段落主题形成主题关键词集，根据主题重要性等因素筛选篇章关键词，解决 TextRank 忽略文本主题的问题。模型主要创新点如下：

a)在词图构建过程中，通过句法分析引入语法信息，通过词嵌入引入语义信息，改善 TextRank 特征单一、结果过于偏向高频词、没有考虑词语法语义的问题；

b)以段落作为抽取关键词的基本单元，减小了词图计算的复杂度，增强了词图内部主题相关度，改善原始 TextRank 对长文本效果不佳的问题；

c)根据段落主题对段落聚类形成主题关键词集，基于主题重要性等因素筛选篇章关键词，解决 TextRank 没有考虑文本主题的问题。

实验证明，模型的准确率 P 、召回率 R 及 F_1 值相比原始 TextRank 及文中所述其他组合均有显著提升。

1 相关工作

关键词抽取是文本处理的重要基础任务之一，自 Luhn[1] 提出基于词频的关键词抽取，学者们提出了许多抽取方案，根据使用的语料可将其分为有监督抽取和无监督抽取。

有监督方法采用分类或序列标注的方式抽取关键词。常用分类器包括朴素贝叶斯、支持向量机、条件随机场、多层感知机等；使用序列标注时多利用神经网络完成。方法效果较好，但需标注语料支持，效果与训练语料相关，应用有较多条件限制。

无监督方法通过量化表示词的重要度抽取关键词，无须标注语料并具有较高普适性，分为基于统计的方法、基于主题模型的方法和基于图的方法。基于统计的方法以统计信息衡量词重要性，对行文敏感且忽略了词的语义关联；基于主题模型的方法以主题划分词类并以词类的中心词作为关键词，虽然考虑了主题因素但主题分布和词类受语料影响大，词类中心词与文本关键词存在一定偏差；基于图的方法将词视为节点，以边表示词间关联，通过计算节点权重抽取关键词，其代表为 TextRank 算法[2]。但 TextRank 算法仅利用了词的

收稿日期：2022-02-24；修回日期：2022-03-31

作者简介：王昊(1994-)，男，黑龙江齐齐哈尔人，硕士研究生，主要研究方向为人工智能、篇章信息处理；刘丹(1969-)，男(通信作者)，四川成都人，副教授，硕导，博士(liudan@uestc.edu.cn)；刘硕(1997-)，男，天津，硕士研究生，主要研究方向为人工智能、信息处理.

共现信息，节点权值受词频影响过大，为此研究者们提出了众多改进模型。

最常见的改进是在 TextRank 中引入统计特征。孙福权[3]等利用万有引力模型综合考虑词的影响力、距离和共现，构建了新的转移概率实现；夏天等[4]定义了词覆盖、词位置、词聚类三种影响力对转移矩阵加权；孟彩霞等[5]根据词在文本中首次出现和最后出现的距离定义了词跨度，并结合词位置对转移矩阵加权；艾金勇[6]则综合考虑词的位置、词性以及词分布修改转移矩阵的权重；Biswas 等[7]从图的结构出发，得出节点权重主要取决于频率、中心性、邻居节点位置等参数；牛永杰等[8]从词出发，得出节点权重的主要影响因素包括词覆盖度、词长、词频、词跨度及词位置；李志强等[9]以词 TF-IDF 值和信息熵的均值为转移概率构建转移矩阵；Mao 等[10]则使用归一化谷歌距离计算词对权重，并引入 WordNet 补充词信息。但统计特征受文本影响大，且上述改进均忽略了词的语义、语法信息，未考虑主题对关键词的影响。

为此部分改进通过组合 TextRank 与其他模型提升效果，组合的模型主要为主题模型和词表示模型。融合主题模型时，部分研究基于主题对候选关键词聚类，基于词聚类和文本信息构建词图进行计算，其代表为 TopicRank[11]、Topical PageRank[12]与 Multipartiterank[13]；另一部分研究则根据主题影响力或主题下词语的相似度对转移矩阵加权[14,15]。融合词表示模型时，主要利用词表示中的语义信息优化转移矩阵。如余本功[16]等基于 Word2Vec 以向量相似度衡量词的语义距离，并综合部分统计信息对转移矩阵加权；夏天[17]则利用词向量对词进行聚类以改进节点间转移概率的计算；Wang 等[18]针对局部信息对全局代表性弱的问题，引入 Doc2Vec 模型以文本向量指引关键词抽取。但上述改进忽略了词的语法信息，使用主题模型时也未考虑文本主题分布对关键词的影响。

2　S-TAKE 模型

本文提出一种基于句法分析与篇章主题的篇章关键词抽取模型 S-TAKE。模型以段落作为抽取关键词的基本文本单元，由段落至篇章分两阶段抽取篇章关键词，包括“段落关键词抽取”及“篇章关键词筛选”两部分。

对于篇章 D，获取其段落集合 $\{P_1,P_2,\cdots,P_n\}$；首先根据“段落关键词抽取算法”构建段落词图 $G_{P_i}=\langle V_{P_i},E_{P_i}\rangle$ 与转移矩阵 C_{P_i}，计算各节点权重并根据权重大小获取段落关键词集 KW_{P_i}；然后利用段落文本生成段落主题向量 T_{P_i}，根据“篇章关键词筛选算法”对段落按主题进行聚类，综合段落关键词形成主题关键词集 KW_{T_i}，依据主题重要度 I_{T_i}、词频等因素对关键词进行筛选得出篇章关键词集合 KW_D。模型原理如图 1 所示。

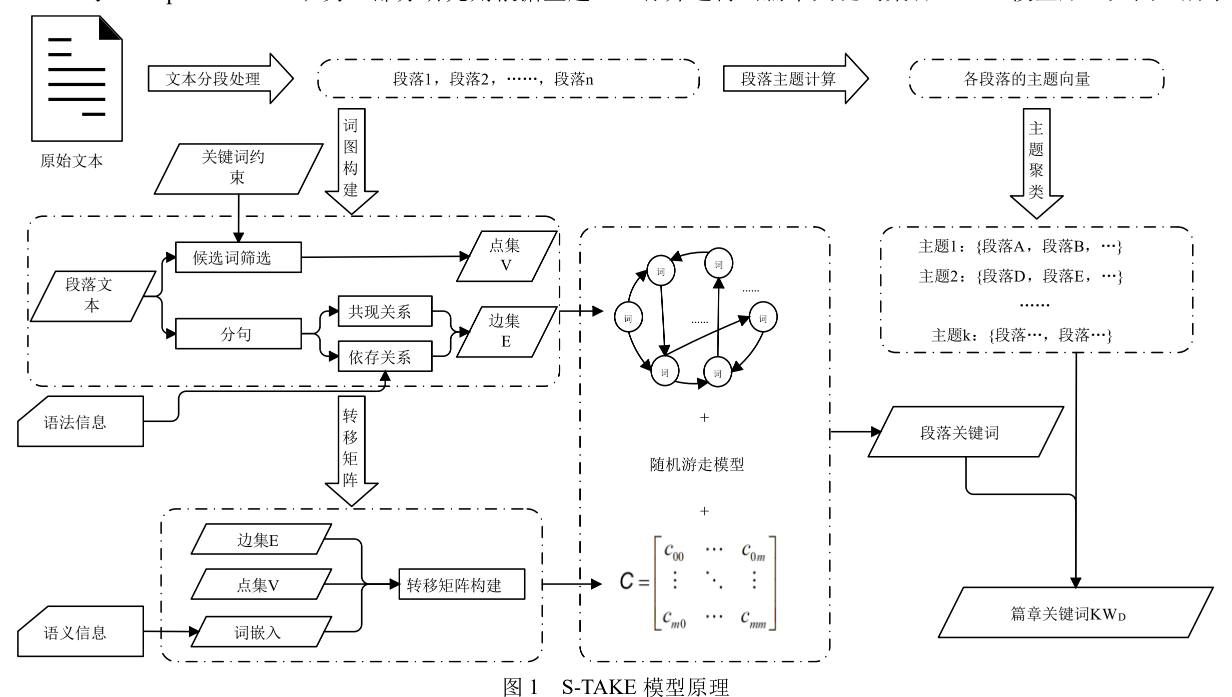


图 1　S-TAKE 模型原理

Fig. 1　Principle of S-TAKE model

2.1　段落关键词抽取算法

篇章通常包含多个主题，传统关键词抽取方法利用整篇文档构建词图，忽略了多主题特征导致词图内主题不统一，对篇章抽取效果不佳。段落作为组成篇章的基本单位，具有段内主题高度一致的特性，且篇章关键词包含于各段的关键词中，因此提出以段落作为获取关键词的基本文本单元。

模型以 TextRank 为基础抽取段落关键词。首先构建段落词图 $G=\langle V,E\rangle$，点集 V 通过对段落文本的分词结果进行筛选获取，边集 E 通过句法分析获取词的语法信息，综合词的语法与共现信息获取；然后利用词嵌入模型获取词的语义信息，根据语义相似度对边赋以不同权重，形成转移矩阵 C；最后利用 PageRank 的计算公式，结合词图结构与转移矩阵计算各节点的权重 PR_{v_i}，根据节点权重的大小获取关键词，实现段落关键词抽取。

2.1.1 基于句法分析的词图构建

词图 $G=\langle V,E\rangle$ 由点集 $V=\{v_1,v_2,\cdots,v_n\}$ 与边集 $E=\{e_1,e_2,\cdots,e_m\}$ 构成，点集 V 对应各候选关键词，边集 E 对应存在关联的候选关键词对。

1)　点集 V 的获取

词图的点对应文本中的词。由于关键词的性质和中文的行文习惯，构建点集 V 时需要对分词结果进行过滤。对于明确的非关键词的过滤可以缩小词图规模，提升词图构建的质量，优化后续的抽取效果。

关键词体现篇章主题，其一定是具有实际意义的实词，因此过滤操作主要根据词性和停用词表进行。模型以名词、动词、数词、形容词、副词等词性作为候选关键词的可能词性，过滤掉其他词性的词及停用词表中的词形成候选关键词集，即词图的点集 V。

2)　边集 E 的获取

词图的边 $e=\langle v_i,v_j\rangle$ 代表其端点 v_i,v_j 对应的词 w_p,w_q 存在关联。TextRank 以词的上下文特征(即词共现)作为衡量词是否存在关联的标准，特征维度单一且受行文影响较大。除上下

文特征外，词语的特征还包括与行文无关的语法信息。词的语法信息通过词间的依存关系体现，一般通过句法分析获取并以三元组 $d=\{w_i,w_j,r_k\},w_i,w_j\in S,r_k\in R$ 表示。

w_i,w_j 为存在依存关系的词，关系由 w_i 指向 w_j；r_k 为弧值，表示依存关系的类型；S 为分析的语句；R 为依存关系类型的集合。当词间存在依存关系且词均属于候选关键词集时，则视为对应顶点间有边相连，若边集 E 中不存在该边，则将这条边添加至边集，即：

$$\begin{aligned}&\text{if } \langle w_p,w_q\rangle\in D_{S_i} \text{ and } \left(v_{w_p},v_{w_q}\in V\right) \text{ and } \left(\langle v_{w_p},v_{w_q}\rangle\notin E\right)\\ &\text{then add } \langle v_{w_p},v_{w_q}\rangle\to E\end{aligned} \tag{1}$$

通过句法分析获取的边体现了词的语法关联，对于行文变化有较强鲁棒性。且语法关联不受词距离影响，能体现远距离的词语关系。但一句话仅包含(词个数-1)条依存关系，在进行词过滤的前提下，通过句法分析获取的边的数量进一步减少，仅使用句法分析构建词图会导致边过于稀疏；同时句子的核心一般为动词，仅使用句法分析得到的边会过分突出动词重要性。

因此构建词图时，模型同时考虑词的语法与共现两个维度的信息，对通过两者获取的边集进行取并操作，提出一种融合词的语法信息和共现信息的词图构建算法，算法实现如下：

算法 1　基于句法分析的词图构建算法

输入：段落文本 P；

输出：段落 P 对应的段落词图 G_P；

a) 初始化词图 G_P，G_P=<V,E>，V=∅，E=∅；

b) 初始化变量 len(滑动窗口 SW)=w；

c) 对 P 分句得句列表{$S_1,S_2,\cdots,S_n$}；

d) FOR i=1 to n:

e) 　对 S_i 分词得词列表{$w_{i1},w_{i2},\cdots,w_{im}$}；

f) 　初始化去除过滤词的语句 S_V=∅；

g) 　FOR w_{ij} in S_i:

h) 　　IF w_{ij}∈过滤词典：

i) 　　　添加 w_{ij}→S_V；

j) 　　　IF w_{ij} 不属于点集 V then 添加 v_{wij}→V；

k) 　获取句子的依存关系集合 D={$d_{i1},d_{i2},\cdots,d_{i(m-1)}$}；

l) 　FOR d_{ij} in D:

m) 　IF V_{wp},V_{wq}∈点集 V and e=<V_{wp},V_{wq}>不属于边集 E:

n) 　　添加 V_{wp},V_{wq}→E；

o) 　FOR j=1 to len(S_V):

p) 　　FOR k=1 to w:

q) 　　　IF $v_{wj},v_{w(j+k)}$不属于边集 E then 添加<$v_{wj},v_{w(j+k)}$>→E；

此时生成的词图同时考虑了词的语法关系和前后词序上的共现关系，解决了 TextRank 没有考虑语法信息、忽略长距离词语关联的问题，避免了单纯使用依存句法构建词图导致词图稀疏与偏重动词的问题。

2.1.2 基于语义加权的转移矩阵构建

转移矩阵是模型获取段落关键词时的另一核心要素，其元素代表不同节点间的转移概率，概率可以利用边的权重的比值表示。TextRank 对各边赋以相同的权重，即从一个节点转移至与其相连的各节点的概率相同，但实际上这种转移具有其倾向性。词图 G 的点对应文本中的词，不同的边关联的词语不同，故可以通过衡量边所关联的词语的关系对不同的边赋以不同的权重。

衡量词语关系最直接的方式就是根据词的语义信息计算其语义距离，词的语义信息一般通过词向量体现，常用词向量包括以 Word2Vec 为代表的静态词向量和以 Bert 为代表的动态词向量，因此利用词向量引入语义信息对转移矩阵进行加权。

以矩阵 $\boldsymbol{C}$ 表示转移矩阵，元素 c_{ij} 表示节点 v_i 到节点 v_j 的转移概率。首先根据词图 G 构建初始转移矩阵 $\boldsymbol{C_0}$：

$$\boldsymbol{C_0}=\begin{bmatrix}c_{00}&\cdots&c_{0m}\\ \vdots&\ddots&\vdots\\ c_{m0}&\cdots&c_{mm}\end{bmatrix} \tag{2}$$

$\boldsymbol{C_0}$ 的横、纵轴对应词图 G 的节点，根据边集 E 初始化 c_{ij}，节点间存在边时有 $c_{ij}=1$，否则有 $c_{ij}=0$。

方法以段落为基本单位构建词图，段落具有较强的主题内聚性，一个段落只对应一个主题，同主题的关键词语义较为接近。因此衡量转移概率时，词的语义越相似对应转移概率越高。使用向量表示词时，常利用向量的余弦距离衡量词语义的远近，公式如下：

$$s_{ij}=\frac{x_{w_i}\cdot x_{w_j}}{\|x_{w_i}\|\|x_{w_j}\|} \tag{3}$$

x_* 为词向量，s_{ij} 为对应的余弦距离，取值为[-1,1]。s_{ij} 越大则向量越相似，词的语义越接近，反之则语义含义越远。考虑词语义信息的同时，还要考虑边出现次数包含的信息。边出现的次数代表着边相关的词关联的次数，关联次数越多，对应词在当前篇章的语境下相关度越高。根据不同词对的余弦距离与出现次数构建权值矩阵 W：

$$\boldsymbol{W}=\begin{bmatrix}w_{00}&\cdots&w_{0m}\\ \vdots&\ddots&\vdots\\ w_{m0}&\cdots&w_{mm}\end{bmatrix} \tag{4}$$

$$w_{ij}=\sum_{\text{词对次数}}s_{ij} \tag{5}$$

利用权值矩阵对初始转移矩阵加权，即可得实际的转移矩阵 C：

$$\boldsymbol{C}=\boldsymbol{C_0}\times\boldsymbol{W} \tag{6}$$

算法 2　基于语义加权的转移矩阵生成算法

输入：段落文本 P，词图结构 G_P。

输出：对应的转移矩阵 C。

a) 以 G_P 的点集大小|V|构建两个|V|*|V|的矩阵，分别为初始转移矩阵 C_0 与权重矩阵 W；

b) 根据 G_P 的边集 E 初始化 C_0；

c) 对 P 分句得句列表{$s_1,s_2,\cdots s_n$}；

d) FOR i=1 to n:

e) 　对 S_i 分词得词列表{$w_{i1},w_{i2},\cdots,w_{im}$}；

f) 　FOR e in 句 S_i 包含的边：

g) 　　获取边关联节点 V_p,V_q 对应词 w_i,w_j 的向量表示 x_{wi},x_{wj}；

h) 　　根据 x_{wi},x_{wj} 计算对应边的权重 S_{ij}；

i) 　　在权重矩阵的对应元素 w_{pq} 与 w_{qp} 上加上权重 S_{ij}；

j) 将初始权重矩阵 C_0 与权重矩阵 W 按位相乘，得到转移矩阵 c；

此时转移矩阵 C 同时考虑了词的语义关联和词对出现的次数信息，得出的转移矩阵更符合中文表达的实际情况。

2.1.3 PR 值与关键词选择

得到词图 G 与转移矩阵 $\boldsymbol{C}$ 后即可利用 PageRank 提出的 PR 值公式计算各节点权值，计算公式如下：

$$PR_{v_i}=(1-d)+d\times\sum_{v_j\in In(v_i)}\frac{c_{ji}}{\sum_{v_k\in O(v_j)}c_{jk}}\times PR_{v_j} \tag{7}$$

PR_{v_i} 表示节点 v_i 的权值；d 为阻尼系数；$In(v_i)$ 表示指向节点 v_i 的节点集合；$O(v_j)$ 表示 v_j 指向的节点集合；c_{ji} 表示转移矩阵 $\boldsymbol{C}$ 中由节点 v_i 转移至节点 v_j 的概率。节点权值需迭代计算至数值稳定，每轮迭代需同时更新所有节点的权重，因此计算过程中采取矩阵运算。用列向量 R_t 表示 t 时刻所有节点的 PR 值向量，则 $t+1$ 时刻的计算公式如下：

$$R_{t+1}=\frac{1-d}{m}\times 1+d\times C\times R_t \tag{8}$$

C 为转移矩阵，m 为词图包含的节点数，迭代运算至权重平稳或达到一定次数后即可停止运算，权重平稳时有 $R_{t+1}=R$。R 为最终的 PR 值矩阵，包含各节点最终的 PR 值，按照 PR 值对节点降序排列，即可选取排名前 K 的词作为输入的关键词。

2.2　基于主题的段落聚类与关键词筛选

原始 TextRank 和各种基于其的改进方法均以篇章为单位构建词图，破坏了原本的文本结构和主题结构，忽略了篇章子主题的信息。中文篇章的主题通常以层次结构体现，一个段落通常只阐述一个主题。主题越重要对应文字描述越多，即对应的段落越多。

因此在获取的段落关键词基础上，模型提出了基于主题的篇章关键词筛选算法。算法首先根据段落 P_i 的文本生成段落主题向量 T_{P_i}，基于段落主题向量对段落进行主题聚类；融合同主题段落的关键词列表形成主题关键词列表；充分考虑文本结构和主题信息，解决原始 TextRank 和各种改进方法忽略文本结构和主题结构的问题；最终根据词频、主题重要度 I_{T_i} 等对主题关键词进行筛选，获取篇章关键词 KW_D。

算法 3　基于主题聚类的篇章关键词筛选算法

输入：段落 P_i 的文本，段落关键词集 KW_{P_i}；

输出：篇章关键词集 KW_D；

a) FOR i=1 to count(P_i):

b)　根据段落 P_i 的文本，生成段落主题向量 T_{P_i}；

c) 根据 T_{P_i} 对段落按主题进行聚类，形成主题集合{$T_1,T_2,\cdots,T_m$}；

d) 合并同主题段落的段落关键词，形成主题关键词集合；

e) FOR i=1 to m:

f)　根据主题对应的段落个数，计算主题重要度 I_{T_i}；

g)　对 KW_{T_i} 中的词，按词在该主题对应段落中的词频降序排列；

h)　取前 $K\times I_{T_i}$ 个关键词，加入篇章关键词集 KW_D；

i) IF count(KW_D)<K:

j)　对所有剩余的主题关键词，按篇章中的词频降序排列；

k)　取前 K-count(KW_D)个不在 KW_D 中的关键词加入 KW_D；

首先使用 Sentence-Transformer 构建各段落的嵌入表示。Sentence-Transformer 基于 Bert 模型，对输入文本的长度存在限制，当输入长度超过限制时，采用截断的方式处理超出限制的文本。

以得到的嵌入表示作为段落的主题向量 T_{P_i}，使用 K-means 算法对各段落的主题向量聚类，形成基于主题的段落集合。由于篇章主题一般不会过多，因此模型对 K-means 的 K 取值为 3。

合并同“主题”下的段落关键词列表，形成主题关键词列表 KW_{T_i}。统计主题关键词列表中各关键词在该主题对应的段落中的出现次数，出现次数越多则该关键词对该主题越有代表性，根据词频对段落关键词列表降序排列。

不同主题对文本的重要程度不同，主题对应的段落越多则该主题越重要，在篇章关键词列表中占比越大，故根据主题对应的段落个数对主题赋权，形成主题权重 I_{T_i}：

$$I_{T_i}=\frac{count(P_j\in T_i)}{count(P_k\in D)} \tag{9}$$

$count(\bullet)$ 表示对括号内元素计数。根据权值，选取每个主题前 $I_{T_i}\times K$ 个关键词作为该主题提供给篇章的关键词，对重复的关键词进行合并，并在剩余的关键词中根据词频选取词语进行补充，形成篇章关键词列表 KW_D。

3　实验数据及分析

3.1　实验数据与环境

实验选取了两个原始数据集，并对其进行筛选构成了实验所用数据。

原始数据集 1 为夏天等人[17]构建的南方周末新闻数据集。随机抽取 300 篇长度在 1000 字以上的文章，并对原始关键词按基本词进行拆分形成 nz_news 数据集，其含有 1090 个未拆分关键词和 1467 个拆分关键词，平均每篇包含 2766.790 个字符，3.633 个未切分关键词和 4.890 个切分关键词。

原始数据集 2 为从各门户网站爬取的新闻数据集，该数据集的关键词为不可拆分的词。随机抽取 300 篇长度在 500-1000 字的文章形成 random_news 数据集，其含有 4642 个关键词，平均每篇包含 729.197 个字符和 15.473 个关键词。

具体实验数据样例如图 2 所示。

{"content": "4月19日，上海海事法院依法扣押了商船三井株式会社所有的、停泊于浙江省舟山市嵊泗马迹山港的226434吨“BAOSTEEL EMOTION”货轮，引发日本政坛关注。\n据《环球时报》报道，日本内阁官房长官菅义伟4月21日称，中国法院对日本商船三井株式会社涉华诉讼采取强制执行措施，日本政府对此表示遗憾。他称，中方此举可能影响两国关系，违背《中日联合声明》中有关放弃战争赔偿的精神，并影响日本企业在中国的投资。\n对此，中国外交部发言人秦刚21日在例行记者会上表示，该案是一起普通商事合同纠纷案。与中日战争赔偿问题无关。中国政府坚持和维护《中日联合声明》各项原则的立场没有变化。中方将继续依法保护外国在华投资企业的合法权益。\n此外，秦刚也在当天会上就日本首相安倍晋三21日向靖国神社供奉祭品一事表示中方已向日方提出交涉，表明了中方的严正立场。\n法院或将依法处理被扣押的船舶。\n据《参考消息》援引日本共同社报道，1936年，侵华战争爆发一年前，日本海运株式会社(现为商船三井株式会社)向中国中威轮船公司租赁了两艘船，合同期为一年。然而，这两艘船从未归还，后来在海上沉没。\n中威轮船公司创始人的孙子向商船三井株式会社提起了诉讼。\n据上海海事法院官网19日晚发布消息对该案进行了情况通报。\n1988年12月30日，原告中威轮船公司、陈震、陈春等为与被告商船三井株式会社定期租船合同欠款及侵权赔偿纠纷一案向上海海事法院提起诉讼，追索“顺丰”轮、“新太平”轮船舶租金及经济损失。上海海事法院对该案进行了公开审理，2007年12月7日，依法作出判决，被告商船三井株式会社支付及赔偿原告陈震、陈春“顺丰”轮和“新太平”轮租金、营运损失、船舶损失及孳息2916477260.80日元（约合人民币2亿元）。\n2010年8月6日，中华人民共和国上海市高级人民法院作出维持原判的终审判决。2010年12月23日，中华人民共和国最高人民法院裁定驳回被告的再审申请。\n通报称，上述案件是一起涉外商事案件，该案判决生效后，原告方依据法律规定，向上海海事法院提出强制执行申请，要求被告履行判决确定的支付和赔偿义务，依法支付迟延履行期间的债务利息。\n上海海事法院于2011年12月28日依法向被执行人商船三井株式会社发出《执行通知书》。期间，双方当事人曾多次进行和解协商未果。为此，上海海事法院依法对被执行人所有的“BAOSTEELEMOTION”轮予以扣押。\n通报称，如商船三井株式会社仍拒不履行义务，法院将依法处理被扣押的船舶。\n日媒称中日关系恶化所致\n前述《参考消息》引据日本新闻网4月20日报道，日本各大媒体当天下午均在自己的网站上报道了上海海事法院扣押日本商船三井株式会社一艘轮船，作为赔偿原中国中威轮船公司在二战期间遭受的财产损失的消息。\n报道援引日本时事通讯社发自北京的评论说，因为战时的财产损失而扣押日本企业在中国国内的现有财产，是极为罕见的事例。其背后是因为日本首相安倍晋三参拜靖国神社等问题而导致的中日关系恶化，中国政府将此作为打压日本的一个重要手段。\n据共同社报道，中国最近发起了一连串针对日企的、与战时被迫赴日劳工有关的索赔诉讼。\n在这些案件中，原告胜诉、被告败诉的裁决可能会导致被告的在华资产进一步被没收。\n针对中方扣押日本船舶一事，日本政府20日开始加紧开展信息收集工作，力图搞清中方意图并冷静应对。\n日本政府相关人士指出：“此案属民事诉讼，政府作出过度反应或有不妥。”该人士表示，今后首先将切实收集相关情报。", "keywords": ["外交部", "日本", "安培晋三", "靖国神社", "商船三井株式会社"], "split_keywords": ["外交部", "日本", "安培晋三", "靖国神社", "商船", "三井", "株式会社"]}

(a)nz_news 样例

{"content": "北京时间5月15日，西部决赛勇士主场113：111战胜马刺，第三节帕楚利亚废小卡的动作成为了本场比赛的绝对焦点，第三节还剩7分55秒时，在面对扎扎-帕楚里亚后撤步投篮后伦纳德左脚再次踩到帕楚里亚脚上，被帕楚利亚的那次犯规，导致他被迫伤退，这也为勇士大逆转吹响了反击的号角。但笔者惊讶的发现，帕楚利亚想要废掉小卡已经不是一天两天的事情了，他们的故事还要追回到2016年2月6日，那时的帕楚利亚还在小牛。那一天，小牛坐镇主场迎来了马刺的挑战，多点开花的马刺上半场就建立起36分的巨大领先优势，下半场小牛也没能扳回比分，最终马刺以116-90轻松取胜。本是一场毫无悬念的比赛，却因为帕楚利亚上演全武行让这场比赛成为当时的焦点。比赛中，帕楚利亚在一次争抢地面球的时候，挽住小卡的胳膊，如果不是小卡身手灵活恐怕他的生涯那时就已经结束了。事情不止这么简单，随后帕楚利亚在一次上篮中，小卡上前防守，帕楚利亚对准小卡的脖子就是一肘直接将小卡击飞。那场比赛不止面对小卡，再一次倒地时，丹尼-格林拿到球后，倒地的帕楚利亚却用脚伸向皇阿玛试图夹住他的脖子。看来帕楚利亚似乎从一开始就那么的不喜欢马刺，即便是老实的小卡也难逃厄运，而且一直被帕楚利亚追着废，一个两次可能是偶然，但永不放弃废小卡的精神，恐怕帕楚利亚真的不能再说是不小心的了吧。", "keywords": ["决赛", "帕楚里亚", "伦纳德", "丹尼-格林", "反击", "比赛", "扎扎", "帕楚利亚", "马刺", "勇士"]}

(b)random_news 样例

图 2　实验数据样例

Fig. 2　Experimental data samples

具体实验环境如表 1 所示。

3.2　方案与指标

实验采取准确率 P、召回率 R 及 F 值 F_1 作为抽取方法效果的评判标准。以 K_A 表示测试数据集提供的正确关键词集合，K_E 表示抽取的关键词集合，各评价指标的计算公式如下：

$$P=\frac{|K_A\cap K_E|}{|K_E|},R=\frac{|K_A\cap K_E|}{|K_A|},F_1=\frac{2\times P\times R}{P+R} \tag{10}$$

实验采用的抽取方法包括：

M_1：原始 TextRank；

M_2：结合句法分析与词共现，以篇章为单位构建词图，

转移矩阵构建与 M_1 相同；

M_3：词图构建与 M_2 相同，以词对出现次数进行加权构建转移矩阵；

M_4：词图构建与 M_2 相同，以 Word2Vec 词向量衡量语义距离，综合语义距离与词对出现次数构建转移矩阵；

M_5：结合句法分析与词共现，以段落为单位构建词图，转移矩阵构建与 M_4 相同，根据词频从所有段落关键词中抽取前 K 个词作为篇章关键词；

M_6：词图构建与转移矩阵构建与 M_5 相同，采用基于主题聚类的篇章关键词筛选算法筛选关键词(即 S-TAKE)。

表 1　实验环境说明

Tab. 1　Experimental environment description

项目	版本或型号
CPU	Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz
内存	8G
系统	CentOS Linux release 7.5.1804(Core)
Python 版本	v3.7
IDE	Pycharm2020.2.3
LTP 库	ltp_data_v3.4.0

3.3　结果与分析

实验一　不同方法在不同关键词个数下抽取情况优劣

为验证各方法抽取关键词效果的优劣，以及抽取不同数量的关键词对结果的影响，分别在 nz_news 与 random_news 数据集上采用不同方法及不同关键词抽取数量进行实验。

实验以 3 为共现窗口长度，分别使用方法 M_1-M_5 抽取 3、5、7 个关键词并与标准答案(拆分关键词)进行对比，实验结果如表 2、3 所示。

表 2　各算法在 nz_news 上的结果对比

Tab. 2　Comparison of the results of Approaches on nz_news

Approach	TopK=5			TopK=7			TopK=10		
	P	R	F1	P	R	F1	P	R	F1
M1	0.291	0.298	0.295	0.243	0.348	0.286	0.200	0.408	0.268
M2	0.320	0.327	0.324	0.267	0.383	0.315	0.222	0.454	0.298
M3	0.334	0.342	0.338	0.278	0.398	0.327	0.226	0.461	0.303
M4	0.333	0.340	0.336	0.284	0.407	0.334	0.230	0.471	0.309
M5	0.345	0.353	0.349	0.289	0.414	0.340	0.235	0.480	0.315

表 3　各算法在 random_news 上的结果对比

Tab. 3　Comparison of the results of Approaches on random_news

Approach	TopK=5			TopK=7			TopK=10		
	P	R	F1	P	R	F1	P	R	F1
M1	0.387	0.125	0.189	0.332	0.150	0.207	0.282	0.182	0.221
M2	0.43	0.139	0.210	0.373	0.169	0.232	0.320	0.207	0.251
M3	0.413	0.133	0.201	0.356	0.161	0.222	0.312	0.202	0.245
M4	0.437	0.141	0.213	0.377	0.170	0.235	0.344	0.223	0.270
M5	0.447	0.145	0.218	0.399	0.180	0.248	0.368	0.238	0.289

由上图结果可知，在五种方法中，方法 M_5 具有最好的效果。随着抽取的关键词数量的增加，各方法在 nz_news 数据集上的 R 值逐渐增加、F_1 值逐渐降低，而在 random_news 数据集上 R 值和 F_1 值则同步增加。

这是因为 nz_news 数据集的篇平均关键词数量较少，因此即使 R 值增加 F_1 值也可能降低。而在 random_news 数据集中，篇平均关键词数量较多，当抽取 10 个关键词时还未达到其篇均的 15 个关键词，因此其 R 值与 F_1 值仍能保持同步增加。

实验二　不同滑动窗口长度对关键词抽取结果的影响

共现窗口长度影决定共现对数目，对词图构建有较大影响。为验证滑动窗口长度对结果的影响，使用方法 M_1、M_5，在 random_news 数据集上依次以 2-6 为窗口长度抽取 10 个关键词，结果如图 3 所示。

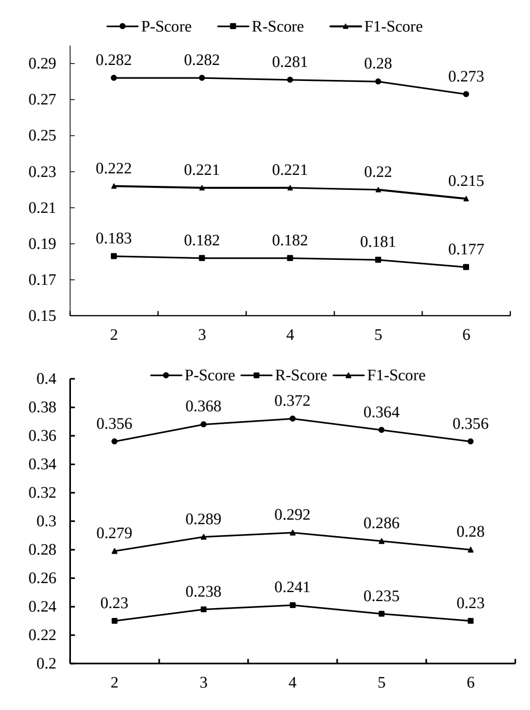


图 3　不同窗口长度下的关键词抽取结果

Fig. 3　Keyword extraction results with different window lengths

方法 M_1 即原始 TextRank 在 random_news 数据集上的抽取效果随着窗口长度增加逐渐降低，符合其原论文使用长度为 2 的共现窗口的结论。方法 M_5 则在共现窗口长度为 4 时取得最好的效果，随后随着窗口长度增加效果逐渐下降。根据结果，推论通过依存关系获取到的信息一定程度上缓解了共现窗口增加时带来的影响。

实验三　验证层次模型的有效性

方法 M_6 即为模型 S-TAKE，为验证其有效性，将其与方法 M_5 进行对比。

由于 random_news 数据集没有段落信息，因此在 nz_news 数据集上进行实验。定义共现窗口长度为 3，分别抽取 5、7、10 个关键词，对比两种算法对于“未拆分关键词”和“拆分关键词”的抽取效果，结果如表 4、表 5 所示。

表 4　“拆分关键词”的抽取结果对比

Tab. 4　Comparison of results for "split keywords"

Approach	TopK=5			TopK=7			TopK=10		
	P	R	F1	P	R	F1	P	R	F1
M5	0.345	0.353	0.349	0.289	0.414	0.340	0.235	0.480	0.315
M6	0.328	0.335	0.332	0.304	0.435	0.358	0.257	0.526	0.346

表 5　“未拆分关键词”的抽取结果对比

Tab. 5　Comparison of results for "none-split keywords"

Approach	TopK=5			TopK=7			TopK=10		
	P	R	F1	P	R	F1	P	R	F1
M5	0.180	0.247	0.208	0.150	0.288	0.197	0.120	0.330	0.176
M6	0.184	0.253	0.213	0.157	0.304	0.208	0.124	0.341	0.182

关键词个数为 5 时，M_6 即 S-TAKE 模型在“未拆分关键词”上的表现效果优于 M_5，但在“拆分关键词”上较低；关键词个数为 7、10 时，S-TAKE 模型的效果则全面优于方法 M_5。

以图 2(a)中截取的语料为例，以“切分关键词”为衡量标准，使用原始 TextRank 抽取 7 个关键词时，其关键词列表为[日本，被告，株式会社，法院，中国，三井，商船，报道，船舶，依法]；使用本文提出的 S-TAKE 方法抽取得到的关键词列表为[日本，安倍晋三，株式会社，中国，商船，三井，靖国神社，依法，船舶，报道]。在排名前 3 的关键词中，原始 TextRank 命中了 2 个，本文算法命中了 3 个；在排名前 7 的关键词中，原始 TextRank 命中了 4 个，本文算法命中了 5

个；在排名前 10 的关键词中，原始 TextRank 命仍旧只命中 4 个，本文算法命中了 6 个。以“未切分关键词为衡量标准时”，在获取 10 个关键词的情况下，原始 TextRank 仅命中了 2 个，本文方法命中了 3 个，对于“复合型”的关键词，两个方法均没能有效识别。

考察语料集给定的关键词，发现可拆分的关键词一般为某主题的细化表达，一般与其主题同时出现在关键词列表中，如“养老金-养老”、“医疗保险-保险”等，且主题词的权重更大不考虑主题且抽取关键词较少时，容易在同一主题下抽取多个词语，即更容易抽取到可拆分的关键词，因此方法 M_5 在抽取词数较少时在“拆分关键词”上效果优于 S-TAKE 模型。但 S-TAKE 模型考虑了主题要素，抽取到了篇章中其他主题的主题词，故其在“未拆分关键词”上的表现效果优于方法 M_5。

4 结束语

本文通过在 TextRank 中引入句法信息与语义信息，提升了算法抽取关键词的能力；同时基于中文行文特点，提出了以段落为基本单位构建词图、根据段落主题聚类获取篇章关键词的思想，解决了 TextRank 忽略文本结构及主题信息的问题。实验结果表明，S-TAKE 模型的效果较原始 TextRank 有显著提高，证明了语法信息与语义信息在关键词获取中的重要作用，证明了主题信息对关键词获取的意义，验证了基于段落主题进行聚类的思想的正确性。

但研究同时提出了新的问题，如何更好的对段落主题进行建模减少误差，如何对不同的依存关系赋以不同的权重，如何对同一条依存边的正反向进行赋权等。后续拟在现有基础上继续研究。

参考文献：

[1] Luhn H P. A Statistical Approach to Mechanized Encoding and Searching of Literary Information [J]. IBM Journal of Research and Development, 1957, 1 (4): 309–317.

[2] Mihalcea R, Tarau P. TextRank: Bringing Order into Text [C]. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. Barcelona, Spain: Association for Computational Linguistics, 2004: 404–411.

[3] 孙福权，张静静，刘冰玉，等. 基于万有引力改进的 TextRank 关键词提取算法 [J]. 计算机应用与软件, 2020, 37 (07): 216-220, 295. (Sun Fuquan, Zhang Jingjing, Liu Bingyu, *et al.* Improved TextRank keyword extraction algorithm based on gravity [J]. Computer Applications and Software, 2020, 37 (07): 216-220, 295.)

[4] 夏天. 词语位置加权 TextRank 的关键词抽取研究 [J]. 现代图书情报技术, 2013 (9): 30-34. (Xia Tian. Study on Keyword Extraction Using Word Position Weighted TextRank [J]. Data Analysis and Knowledge Discovery, 2013 (9): 30-34.)

[5] 孟彩霞，张琰，李楠楠. 基于 TextRank 的关键词提取改进方法研究 [J]. 计算机与数字工程, 2020, 48 (12): 3022-3026. (Meng Caixia, Zhang Yan, Li Nannan. Research on the improvement method of keyword extraction based on TextRank [J]. Computer and Digital Engineering, 2020, 48 (12): 3022-3026.)

[6] 艾金勇. 融合多特征的 TextRank 藏文文本关键词抽取方法研究 [J]. 情报探索, 2020 (07): 1-6. (Ai Jinyong. A Study on TextRank Keyword Extraction Method for Tibetan Texts Incorporating Multiple Features [J]. Information Research, 2020 (07): 1-6.)

[7] BISWAS S K, BORDOLOI M, SHREYA J. A graph based keyword extraction model using collective node weight [J]. Expert Systems with Applications, 2018, 97: 51-59.

[8] 牛永洁，姜宁. 关键词提取算法 TextRank 影响因素的研究 [J]. 电子设计工程, 2020, 28 (12): 1-5. (Niu Yongjie, Jiang Ning. Research on influence factors of keyword extraction algorithm TextRank [J]. Electronic Design Engineering, 2020, 28 (12): 1-5.)

[9] 李志强，潘苏含，戴娟，等. 一种改进的 TextRank 关键词提取算法 [J]. 计算机技术与发展, 2020, 30 (03): 77-81. (Li Zhiqiang, Pan Suhan, Dai Juan, *et al.* An improved TextRank keyword extraction algorithm [J]. Computer Technology and Development, 2020, 30 (03): 77-81.)

[10] Mao Xiangke, Huang Shaobin, Li Rongsheng, *et al.* Automatic Keywords Extraction Based on Co-Occurrence and Semantic Relationships Between Words [J]. IEEE Access, 2020, PP (99): 1-1.

[11] Bougouin A, Boudin F, Daille B. TopicRank: Graph-Based Topic Ranking for Keyphrase Extraction [C]. In: Proceedings of the Sixth International Joint Conference on Natural Language Processing. Nagoya, Japan: Asian Federation of Natural Language Processing/ACL, 2013: 543–551.

[12] Liu Zhiyuan, Huang Wenyi, Zheng Yabin, *et al.* Automatic Keyphrase Extraction via Topic Decomposition [C]. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Massachusetts, USA: Association for Computational Linguistics, 2010: 366–376.

[13] BOUDIN F. Unsupervised key phrase extraction with multipartite graphs [C]// Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT, Association for Computational Linguistics, New Orleans: June 1-6, 2018, 2: 667-672.

[14] STERCKX L, DEMEESTER T, DELEU J, *et al.* Creation and evaluation of large Keyphrase extraction collections with multiple opinions [J]. Language Resources and Evaluation, 2017, 52: 503-532.

[15] 张兵磊. 基于 TextRank 和 LDA 的中文短文本分类研究 [J]. 信息与电脑（理论版）, 2021, 33 (06): 12-14. (Zhang Binglei, Research on Chinese short text classification based on TextRank and LDA [J] , China Computer & Communication, 2021, 33 (06): 12-14.)

[16] 余本功，张宏梅，曹雨蒙. 基于多元特征加权改进的 TextRank 关键词提取方法 [J]. 数字图书馆论坛, 2020 (03): 41-50. (Yyu Bengong, Zhang Hongmei, Cao Yyumeng. Improved TextRank Keyword Extraction Method Based on Multivariate Features Weighted [J]. Digital Library Forum, 2020 (03): 41-50.)

[17] 夏天. 词向量聚类加权 TextRank 的关键词抽取 [J]. 数据分析与知识发现, 2017, 1 (2): 28-34. (Xia Tian, Extracting Keywords with Modified TextRank Model [J] , Data Analysis and Knowledge Discovery, 2017, 1 (2): 28-34.)

[18] Wang Wei, Li Xiangshun, Yyu Sheng. Chinese Text Keyword Extraction Based on Doc2vec And TextRank [C]// 2020 Chinese Control And Decision Conference (CCDC) . 2020.